M16C/62P Group

Operation of Timer A (event counter mode, free run type selected)

1. Abstract

In event counter mode, choose functions from those listed in Table 1. Operations of the circled items are described below. Figure 1 shows the operation timing, and Figure 2 shows the set-up procedure.
A reference program is an example when using the Timer A0 interrupt based on the setting procedure of Figure 2.

2. Introduction

This application note is applied to the M16C/62P group Microcomputers.

This program can be operated under the condition of M16C family products with the same SFR(Special Function Register) as M16C/62P Group products. Because some functions may be modified of the M16C family products, see the user's manual. When using the functions shown in this application note, evaluate them carefully for an operation

3. Choosed functions

Table 1. Choosed functions

Item	Set-up		Item	Set-up	
Count source	O	Input signal to TAiIN (counting falling edges)	Pulse output function	O	No pulses output
					Pulses output
		Input signal to TAiIN (counting rising edges)	Count operation type		Reload type
				O	Free-run type
		Timer overflow (TB2/TAj overflow)	Factor for switching between up and down	O	Content of up/down flag
					Input signal to TAiOUT

Note: j=i-1, but j=4 when i=0

4. Operation

(1) Setting the count start flag to “1” causes the counter to count the falling edges of the count source.
(2) Even if an underflow occurs, the content of the reload register is not reloaded, but the count continues. At this time, the timer Ai interrupt request bit goes to “1”.
(3) If switching from an up count to a down count or vice versa while a count is in progress, the switch takes effect from the next effective edge of the count source.
(4) Even if an overflow occurs, the content of the reload register is not reloaded, but the count continues. At this time, the timer Ai interrupt request bit goes to “1”.

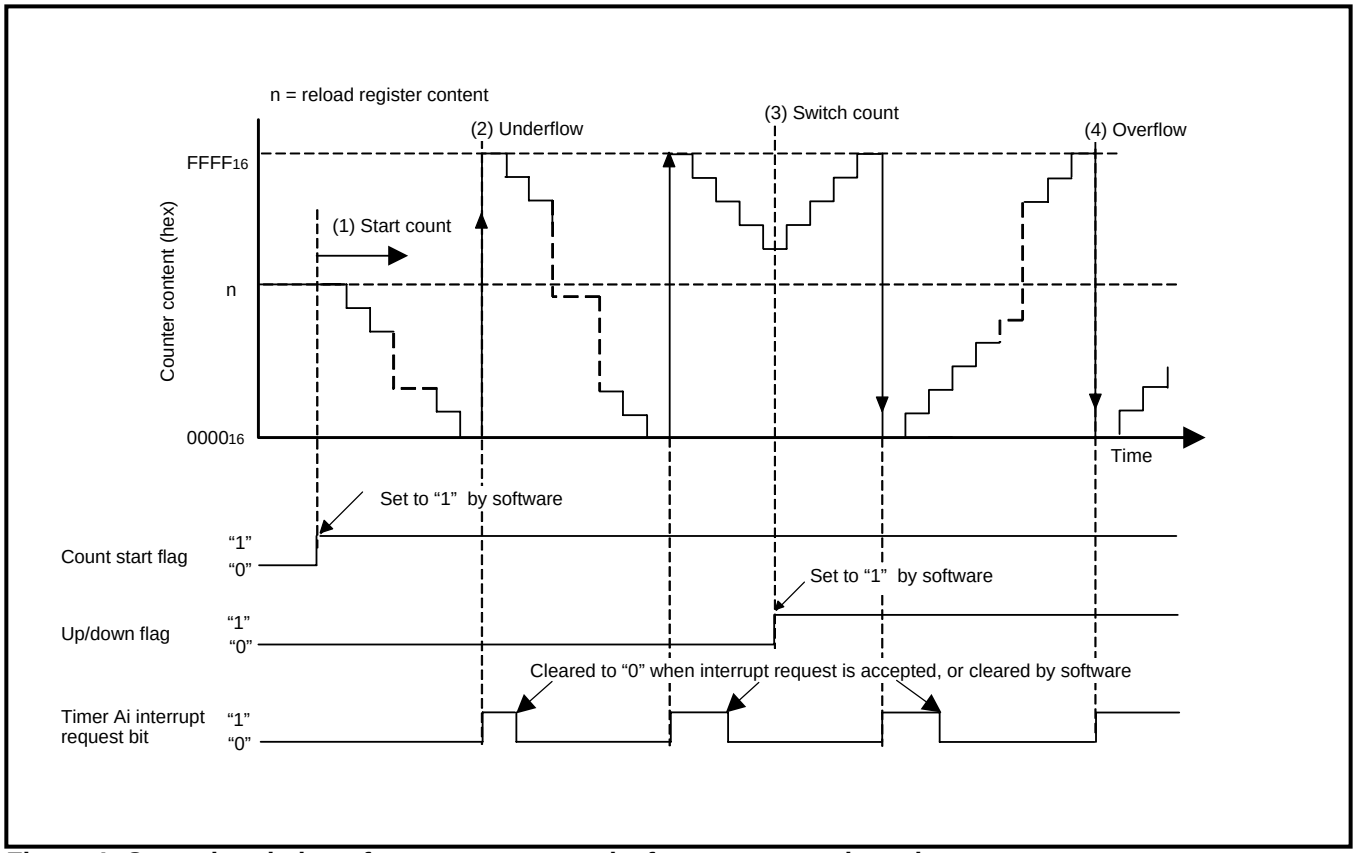

Figure 1. Operation timing of event counter mode, free run type selected

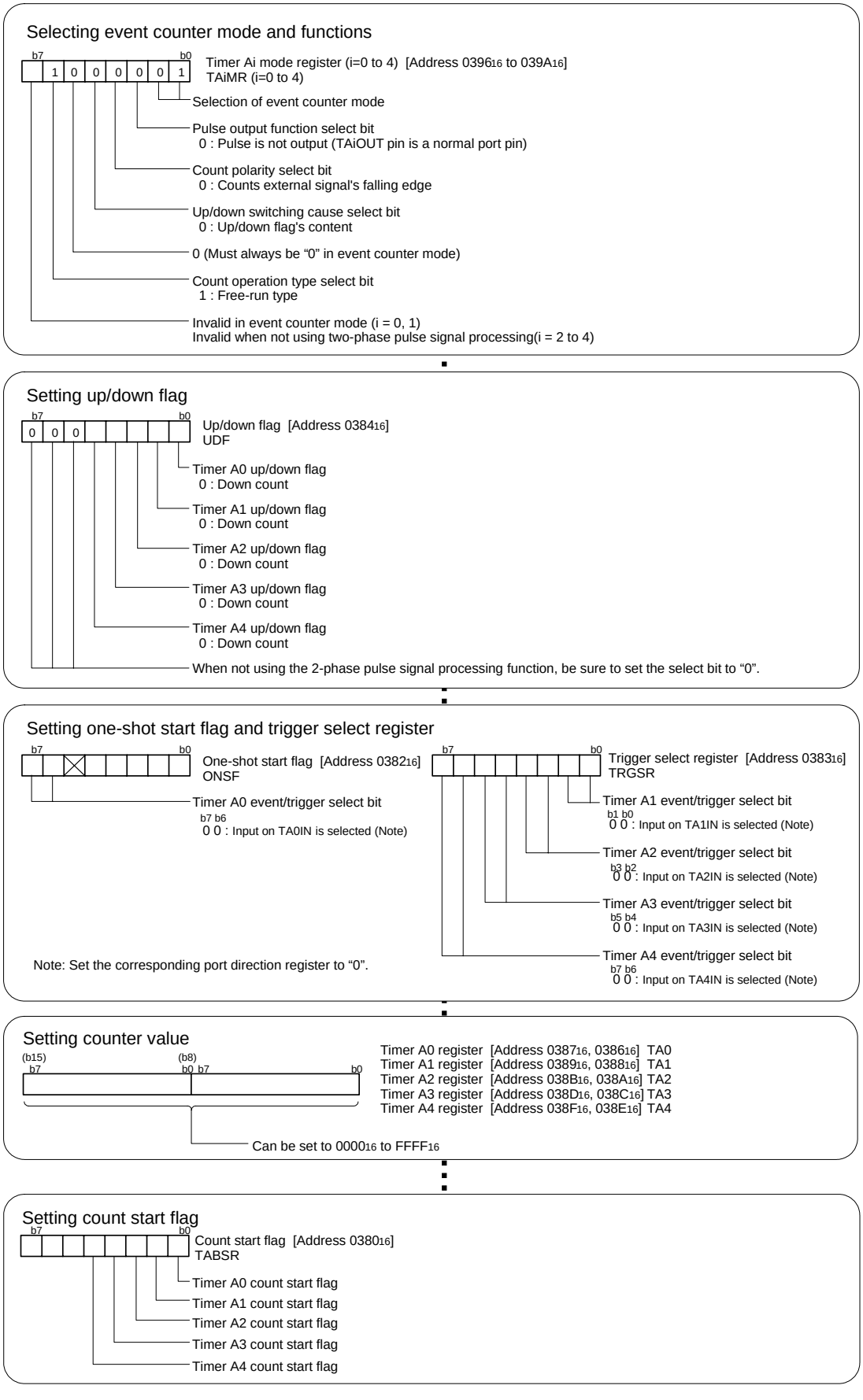
Selecting event counter mode and functions
b7 b0
1 0 0 0 0 0 1
Timer Ai mode register (i=0 to 4) [Address 0396₁₆ to 039A₁₆]
TAiMR (i=0 to 4)
Selection of event counter mode
Pulse output function select bit
0 : Pulse is not output (TAiOUT pin is a normal port pin)
Count polarity select bit
0 : Counts external signal's falling edge
Up/down switching cause select bit
0 : Up/down flag's content
0 (Must always be "0" in event counter mode)
Count operation type select bit
1 : Free-run type
Invalid in event counter mode (i = 0, 1)
Invalid when not using two-phase pulse signal processing(i = 2 to 4)
Setting up/down flag
b7 b0
0 0 0
Up/down flag [Address 0384₁₆]
UDF
Timer A0 up/down flag
0 : Down count
Timer A1 up/down flag
0 : Down count
Timer A2 up/down flag
0 : Down count
Timer A3 up/down flag
0 : Down count
Timer A4 up/down flag
0 : Down count
When not using the 2-phase pulse signal processing function, be sure to set the select bit to "0".
Setting one-shot start flag and trigger select register
b7 b0
One-shot start flag [Address 0382₁₆]
ONSF
Timer A0 event/trigger select bit
b7 b6
0 0 : Input on TA0IN is selected (Note)
b7 b0
Trigger select register [Address 0383₁₆]
TRGSR
Timer A1 event/trigger select bit
b1 b0
0 0 : Input on TA1IN is selected (Note)
Timer A2 event/trigger select bit
b3 b2
0 0 : Input on TA2IN is selected (Note)
Timer A3 event/trigger select bit
b5 b4
0 0 : Input on TA3IN is selected (Note)
Timer A4 event/trigger select bit
b7 b6
0 0 : Input on TA4IN is selected (Note)
Note: Set the corresponding port direction register to "0".
Setting counter value
(b15) (b8)
b7 b0 b7 b0
Timer A0 register [Address 0387₁₆, 0386₁₆] TA0
Timer A1 register [Address 0389₁₆, 0388₁₆] TA1
Timer A2 register [Address 038B₁₆, 038A₁₆] TA2
Timer A3 register [Address 038D₁₆, 038C₁₆] TA3
Timer A4 register [Address 038F₁₆, 038E₁₆] TA4
Can be set to 0000₁₆ to FFFF₁₆
Setting count start flag
b7 b0
Count start flag [Address 0380₁₆]
TABSR
Timer A0 count start flag
Timer A1 count start flag
Timer A2 count start flag
Timer A3 count start flag
Timer A4 count start flag
Start count

Figure 2. Set-up procedure of event counter mode, free run type selected

5. The example of reference program

```
;***********************************************************************************
;
;    M16C/62P Program Collection
;
;    FILE NAME : rjj05b0692_src.a30
;    CPU       : M16C/62P Group
;    FUNCTION  : Operation of Timer A (event counter mode, free run type selected)
;    HISTORY   : 2004.12.24    Ver 1.00
;
;    Copyright(C)2004, Renesas Technology Corp.
;    Copyright(C)2004, Renesas Solutions Corp.
;    All rights reserved.
;
;***********************************************************************************
;***********************************************************************************
;        Include
;***********************************************************************************
;
        .LIST       off                 ;Stops outputting lines to the assembler list file
        .INCLUDE    sfr62p.inc          ;Reads the file that defined SFR
        .LIST       on                  ;Starts outputting lines to the assembler list file
;
;***********************************************************************************
;        Symbol definition
;***********************************************************************************
;
RAM_TOP             .equ    00400h      ;Start address of RAM
RAM_END             .equ    013ffh      ;End address of RAM
ROM_TOP             .equ    0f4000h     ;Start address of ROM
VECT_TOP            .equ    0ffe00h     ;Start address of vect_top
FIXED_VECT_TOP      .equ    0fffdch     ;Start address of fixed_vect_top
SB_BASE             .equ    00380h      ;Base address of sb
;
;***********************************************************************************
;        Program area
;***********************************************************************************
;=================================================================================
;        Start up
;=================================================================================
;
            .section    program,code        ;Declares section name and section type
            .org        ROM_TOP             ;Declares start address
            .sb         SB_BASE             ;
;
START:
```

```
                ldc         #RAM_END+1,isp ;Sets interrupt stack pointer
                ldc         #SB_BASE,sb    ;Sets sb register
;
                mov.b       #03h,prcr      ;Removes protect
                                           ;Set processor mode registers 0 and 1
                mov.w       #0800h,pm0     ;Single-chip mode
                                           ;No expansion, No wait
                mov.w       #2008h,cm0     ;Xcin-Xcout High
                                           ;Xin-Xout High, Main clock is No divison
                mov.b       #0,prcr        ;Protects all registers
;
                ldintb      #VECT_TOP      ;Sets interrupt table register

                mov.w       #0,r0          ;Clears WORKRAM area
                mov.w       #((RAM_END+1)-RAM_TOP)/2,r3
                mov.w       #RAM_TOP,a1
                sstr.w
;
;=============================================================================
;       Main program
;=============================================================================
                mov.b       #01000001b,ta0mr ;Timer A0 mode register
;                               |     ++----------;Event counter mode
;                              +--------------------;Free-run type
                mov.b       #00000000b,udf  ;Up/down flag
;                               | | |     +----------;Down count
;                               +++------------------;When not using the 2-phase pulse signal processing
;                                                  ;function,set the select bit to "0"
                mov.b       #00000000b,onsf ;One-shot start flag
;                               ++--------------------;Input on ta0in is selected
                bclr        pd7_1          ;(Note)Set the corresponding port direction
;                                          ;register to "0"(TA0IN)
                mov.w       #5,ta0         ;Timer A0 register
                mov.b       #00000011b,ta0ic ;Interrupt control register
;                                     |+++----------;Interrupt priority level select bit
;                                     |          ;(011:Level 3, interrupt disabled)
;                                    +----------;Interrupt request bit (0:interrupt not requested)
                mov.b       #00000001b,tabsr ;Count start flag
;                                        +----------;Starts counting
                fset        i              ;Set interrupt enable flag
MAIN:
                jmp         MAIN
;
;=============================================================================
;       Interrupt program
;=============================================================================
TA0_INT:
```

```
;                   ;/ TA0 interrupt routine /
                    reit
;
;=============================================================================
;           Dummy interrupt processing program
;=============================================================================
DUMMY:
                    reit
;
;*****************************************************************************
;           Setting of variable vector table
;*****************************************************************************
;
                    .section   vect,romdata
                    .org                VECT_TOP + (4 * 4)
;
                    .lword              DUMMY           ;INT3 interrupt vector
                    .lword              DUMMY           ;TB5 interrupt vector
                    .lword              DUMMY           ;TB4 interrupt vector
                                                        ;UART1 bus collision detection interrupt vector
                    .lword              DUMMY           ;TB3 interrupt vector
                                                        ;UART0 bus collision detection interrupt vector
                    .lword              DUMMY           ;SI/04/INT5 interrupt vector
                    .lword              DUMMY           ;SI/03/INT4 interrupt vector
                    .lword              DUMMY           ;UART2 bus collision detection interrupt vector
                    .lword              DUMMY           ;DMA0 interrupt vector
                    .lword              DUMMY           ;DMA1 interrupt vector
                    .lword              DUMMY           ;KEY interrupt vector
                    .lword              DUMMY           ;A-D interrupt vector
                    .lword              DUMMY           ;UART2 transmit/NACK interrupt vector
                    .lword              DUMMY           ;UART2 receive/ACK interrupt vector
                    .lword              DUMMY           ;UART0 transmit/NACK interrupt vector
                    .lword              DUMMY           ;UART0 receive/ACK interrupt vector
                    .lword              DUMMY           ;UART1 transmit/NACK interrupt vector
                    .lword              DUMMY           ;UART1 receive/ACK interrupt vector
                    .lword              TA0_INT         ;TA0 interrupt vector
                    .lword              DUMMY           ;TA1 interrupt vector
                    .lword              DUMMY           ;TA2 interrupt vector
                    .lword              DUMMY           ;TA3 interrupt vector
                    .lword              DUMMY           ;TA4 interrupt vector
                    .lword              DUMMY           ;TB0 interrupt vector
                    .lword              DUMMY           ;TB1 interrupt vector
                    .lword              DUMMY           ;TB2 interrupt vector
                    .lword              DUMMY           ;INT0 interrupt vector
                    .lword              DUMMY           ;INT1 interrupt vector
                    .lword              DUMMY           ;INT2 interrupt vector
;
```

```
;******************************************************************************
;       Setting of fixed vector
;******************************************************************************
;
                .section   f_vect,romdata
                .org            FIXED_VECT_TOP
;
                .lword          DUMMY           ;Undefined instruction interrupt vector
                .lword          DUMMY           ;Overflow (INTO instruction) interrupt vector
                .lword          DUMMY           ;BRK instruction interrupt vector
                .lword          DUMMY           ;Address match interrupt vector
                .lword          DUMMY           ;Single-step interrupt vector
                .lword          DUMMY           ;Watchdog timer interrupt vector
                                                ;Oscillation stop and Re-oscillation detection interrupt
                                                ;vector
                                                ;Voltage down detection interrupt vector
                .lword          DUMMY           ;DBC interrupt vector
                .lword          DUMMY           ;NMI interrupt vector
                .lword          START           ;Sets start vector
;
                .end
```

6. Referense

Hardware manual
M16C/62P Group Hardware Manual
(Use the most recent version of the document on the Renesas Technology Web site.)

Technical news/Technical update
(Use the most recent version of the document on the Renesas Technology Web site.)

Web-site and contact for support

Renesas Technology Web site
http://www.renesas.com/en/m16c

Inquiries
http://www.renesas.com/inquiry
csc@renesas.com

Revision

Rev.	Issue data	Revised	
		Page	Point
1.00	2004.12	-	First edition issued
1.10	2006.01	2	Figure 1 modified

Keep safety first in your circuit designs!

1. Renesas Technology Corporation puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage.
   Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corporation product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corporation or a third party.
2. Renesas Technology Corporation assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor for the latest product information before purchasing a product listed herein.
   The information described here may contain technical inaccuracies or typographical errors.
   Renesas Technology Corporation assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
   Please also pay attention to information published by Renesas Technology Corporation by various means, including the Renesas Technology Corporation Semiconductor home page (http://www.renesas.com).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corporation semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corporation is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
   Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corporation for further details on these materials or the products contained therein.